Grammar Correction for Event - to - Sentence

Ethan Tien

LMC 4702

Fall 2018

Faculty Member 1:

Printed: Mark Riedl

Signature:

Faculty Member 2:

Printed: Dhruv Batra

Signature: Dhruv Batra

Grammar Correction for *Event-to-Sentence*

Ethan Tien, Wesley Cheung, Prithviraj Ammanabrolu, Lara J. Martin, Mark O. Riedl
School of Interactive Computing, Georgia Institute of Technology

Abstract

Automated story generation is the problem of generating words and sentences with the goal of telling a coherent story. To date, we have the *Event-to-Event* model that can generate more events that give us the building blocks for the sequence of events in a story. However, the difficulty lies in trying to translate these building blocks back into coherent sentences that tell a story. Although we have a baseline implementation of the *Event-to-Sentence* model, the results still end up being incoherent which leaves much room for improvement. We present a technique that takes advantage of a grammar correction model in order to fix the errors in the output and increase the comprehensibility.

Introduction

Automated story generation (Klein et al. 1973) is a field of study that has continued to be a popular area of research ever since artificial intelligence became a field. This problem involves automatically generating words and sentences with the ultimate goal of telling a coherent story. Most popular story generation implementations require large amounts of human authoring. This involves a human providing many guidelines and restrictions that the target story should follow. Although they perform well in restricted domains, these models cannot be generalized to other topics that may not be included in the human authoring stage.

One way such a model can be implemented is to reduce sentences as *events* (Martin et al. 2018), which are created by extracting basic semantic information from sentences. Currently, work is being done a recurrent encoder-decoder neural network that generates new events from old events, being named as the *Event-to-Event* algorithm. Although this can generate new events to make up a story, these events are not human readable or grammatically correct, which defeats the purpose of creating a coherent story. To address this, work is being done on a new neural network architecture, named *Event-to-Sentence*, which attempts to fill in the missing details of these incoherent events to make them human readable. By having these two networks work together, we can theoretically have the ability to create stories in any conceivable domain without the need for additional human authoring.

Although much work has been done on the *Event-to-Sentence* implementation, the results are still very incoherent and requires lot of improvements. One way we can address the incoherence and plethora of errors of the network's output is to add a layer of grammar correction at the end of the pipeline. Upon further analysis of the output, the most common grammatical errors include incorrect verb conjugation or missing a direct object. By implementing a model that can fix these types of errors, we can make the output of the overall model much more understandable and more closely resemble that of a coherent story.

Background

Grammar correction has been a long standing research topic that people have been trying to solve. However, the previous research in this field is more focused on correcting the errors that humans make, rather than the machine-translation output at hand. The National University of Singapore (NUS) Corpus of Learner English (NUCLE) corpus (Dahlmeier, Ng, and Wu 2013) was created to combat the fact that there was no previously created large data set for grammar correction. This is comprised of 1,414 essays written by people who were learning English as a second language. Then, the mistakes in these essays were annotated with a specific error type and a corrected version of the sentence. Similarly, there exists the publicly available Lang-8 dataset (Tajiri, Komachi, and Matsumoto 2012), that contains a large number of crowd-sourced annotations. Although these are good resources for producing generic grammar correction models, the task at hand is a bit more complex because the sentences are machine generated. Because of this high complexity, these models may produce some questionable output.

Methods

In this section we describe the details of the model that we use to train *Event-to-Sentence*. After collecting and cleaning the training corpus, and running the inputs through the *Event-to-Event* model, we can determine how to optimize the *Event-to-Sentence* in order to generate the best output.

Experimental Setup

The data we use is scraped from science fiction wiki summaries. After clustering and cleaning this dataset, we were able to "eventify" the stories — turning each sentence into events. To construct events, we extract the verb, subject, object, and modifier from a sentence. A level of abstraction is added on top of this in order to help the models train faster and decrease sparsity. More specifically, the events and sentences are *generalized*. Essentially, each identified named entity (Finkel, Grenager, and Manning 2005) is replaced with its named entity category. Also, each noun that was not identified as a named entity is replaced by the WordNet (Miller 1995) Synset two levels up in the inherited hypernym hierarchy, giving us a general category.

Then, we can feed in the generalized events as inputs to the neural network, and have the original sentences as the output. By training a LSTM RNN with these parameters, we are able to have a model that attempts to take in events and fill out the missing pieces to create a fully coherent sentence. We use PyTorch to implement these models, which allowed us to implement a beam search algorithm instead of a greedy search to aid finding a more optimal solution while decoding.

Unfortunately, the output from this model sometimes gives us some nonsensical output, which leads us to attempt to fix up these sentences using some sort of a grammar correction model.

Experiments

We experiment with two different grammar correction models that we found and deemed worthy of experimentation. The first technique is a phrase-based machine translation approach (Junczys-Dowmunt and Grundkiewicz 2016), which was submitted as the state of the art for automatic grammar error correction to EMNLP 2016. The other one was a sequence-to-sequence based approach (Schmaltz et al. 2017) that was trained off of the NUCLE and Lang-8 dataset.

Since the models were pre-trained off of data that represented plain English, and not generalized sentences, for the sake of demonstrating only grammar functionality, we mocked a pseudo slot-filling algorithm, where we converted each Synset into a leaf node in its hypernym hierarchy, and each named entity category into a random noun that exists within said category.

Results and Discussion

In *Table 1*, we show the BLEU and Perplexity evaluation metrics comparing the original *Event-to-Sentence* baseline with the improved pipeline with our proposed grammar correction model. For clarity's sake, we have also attached multiple examples of its performance in *Appendix A*.

From the results in the table, it does not seem like adding any form of grammar correction to the pipeline yields better results. It is to our belief that BLEU is not a very good metric for evaluating this type of task. BLEU emphasizes recreating the input by measuring overlapping n-grams, but this is not the focus of our task. Thus, it makes sense that all of our experiments yield relatively low BLEU scores. Furthermore, perplexity is also a weak measure for this task, as it measures how ”surprised” a model is, and it is hard to see how that helps understanding when generating sentences from *Events*. However, by doing some human evaluation, we can see that the provided examples show that the sentences that went through the grammar corrector make more sense than those that did not (shown in *Appendix A*). The model performs very well when it is given a sentence that can be easily fixed (for example, with just a change of tense or conjugation). However, when it takes in an input that is extremely far from a correct or coherent sentence, it just does some random guessing or doesn't do anything at all. With some manual evaluation, the sequence-based approach seems to perform better, where it attempts to correct more errors than the phrase-based model. Also, because all concepts of punctuation are stripped when training the *Event-to-Sentence* model, the grammar correction models have the beneficial side effect of inserting punctuation and capitalization where necessary.

Table 1: Results of the Grammar Correction (GC) experiments.

Model	BLEU score	Average Perplexity
Event-to-Sentence	**11.81%**	**494.62**
E2S + Phrase-based GC	7.59%	1531.19
E2S + Sequence-based GC	8.28%	1323.61

Conclusion

We introduce a way of improving the original *Event-to-Sentence* model. Originally, the model had trouble generating coherent and readable sentences. However, by implementing and incorporating a grammar correction model to help the sentences make more sense, we are able to have more coherent sentences that will eventually work towards building up a story. Although the incorporation of grammar correction models did not increase performance based on the performance metrics used, we still believe this is a right step forward into making sense of what was previously nonsensical output.

Future Work

Automated story generation still proves to be a tough problem that we have to pick apart and solve step by step. With the introduction of a grammar correction model to the pipeline, we show that create more coherent results that could lead to constructing complete stories. However, as a whole, there are many more ways we can work on improving *Event-to-Sentence*. Wwe can improve the underlying model of *Event-to-Sentence* itself by doing smarter beam searching or using a different kind of dataset. Or, we could replace the entirety of the current *Event-to-Sentence* model and use some sort of template approach to take in *Events* and construct sentences that we already know are coherent.

References

Dahlmeier, D.; Ng, H. T.; and Wu, S. M. 2013. Building a large annotated corpus of learner english: The nus corpus of learner english. In *Proceedings of the eighth workshop on innovative use of NLP for building educational applications*, 22–31.

Finkel, J. R.; Grenager, T.; and Manning, C. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, 363–370. Association for Computational Linguistics.

Junczys-Dowmunt, M., and Grundkiewicz, R. 2016. Phrase-based machine translation is state-of-the-art for automatic grammatical error correction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1546–1556. Austin, Texas: Association for Computational Linguistics.

Klein, S.; Aeschlimann, J.; Balsiger, D.; et al. 1973. Automatic novel writing: A status report. *Wisconsin University*.

Martin, L. J.; Ammanabrolu, P.; Wang, X.; Hancock, W.; Singh, S.; Harrison, B.; and Riedl, M. O. 2018. Event Representations for Automated Story Generation with Deep Neural Nets. In *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, 868–875.

Miller, G. A. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

Schmaltz, A.; Kim, Y.; Rush, A.; and Shieber, S. 2017. Adapting sequence models for sentence correction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2807–2813. Copenhagen, Denmark: Association for Computational Linguistics.

Tajiri, T.; Komachi, M.; and Matsumoto, Y. 2012. Tense and aspect error correction for esl learners using global context. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, 198–202. Association for Computational Linguistics.

Appendix A

Input Event	**Event-to-Sentence**	**E2S + Phrase Based GC**	**E2S + Sequence Based GC**
⟨ Synset('negotiator.n.01'), become-109.1-1, ∅, Synset('frequency.n.01') ⟩	the haggler come to the answering city council	The haggler comes to the answering city council .	The haggler came to the answering city council .
⟨ PRP, try-61.1, amuse-31.1, ∅⟩	he will try to kill he if he attempts to find he	He will try to kill him if he attempts to find his .	He will try to kill him if he attempts to find him .
⟨ Synset('male.n.02'), future_having-13.3, Synset('male.n.02'), ∅⟩	the the clan begin to blackmail biological agent	The family begins to blackmail biological agent .	The clan begins to blackmail biological agents .

Figure 1: This shows some experimental results of the Event-to-Sentence, as well as results after passing that output to our Grammar Correction models. Here, we use a simple slot filler that replaces each named entity with a random noun in that category, and replaces each Synset with a leaf node in the hypernym hierarchy.